University of Colorado Boulder

UNIVERSITY OF COLORADO BOULDER

# Evaluating the Effectiveness of Using Weather Conditions as a Predictor for Occurrence and Duration of Flight Delays


***Student :***
Ella Bronaugh
Mayur Dalvi
Reza Naiman
Tristan Levy-Park

***Teacher :***
Dr. Kris Pruitt

***Domain Experts :***
Dr. Edward Ochoa
Leon Shen
Logan Gage


November 14, 2024

# Contents

# 1 Introduction

Flight delays due to adverse weather conditions pose significant challenges to the aviation industry. Researchers Christopher J. Goodman and Jennifer D. Small Griswold found that extreme weather events were responsible for 32.6% of the total delay minutes recorded in the National Airspace System (NAS) from 2003 to 2015, with severe weather causing up to 82% of delay minutes in some instances [3]. Such delays have an impact on the environment as well as economic performance in the airline industry. [6, 7, 8]. These economic impacts include increasing fuel consumption and emissions. For example, when aircraft are delayed, they often remain idling on the runway or need to reroute, leading to additional greenhouse gas emissions and contributing to air pollution. The annual economic impact of airline delays was estimated by one study to be $31.2 billion in 2010 and $40.2 billion in other estimates [1]. In addition, climate change and air transportation have a reciprocal relationship: aircraft emissions contribute to anthropogenic climate change while atmospheric changes directly impact operations in the airline industry [5]. Not only this–delays due to inclement weather are increasing over time. In comparing the impact of a winter storm in December 2021 versus December 2022, aggregated total passenger "dwell time" in airports saw an increase of approximately 12 million hours [9]. There are several factors contributing to increasing frequency of weather delays and the economic and environmental impacts are substantial; thus, the ability to accurately predict the occurrence and duration of flight delays and manage these disruptions is increasingly crucial. This study aims to answer the following question: does the use of weather conditions as a predictor increase the accuracy of existing models that classify flight delays and allow us to estimate their duration?

Previous studies have highlighted the interaction between meteorological conditions and aviation operations, employing various Machine Learning (ML) and data-driven approaches to forecast delays [2, 3, 4]. These findings emphasized the importance of understanding weather patterns and airport-specific vulnerabilities to optimize flight schedules and improve operational efficiency. A group of researchers, Kerim Kiliç and Jose M. Sallan, examined arrival delays across the United States airport network using a variety of models [4]. Their analysis, based on 2017 flight and weather data, found the Gradient Boosting Machine (GBM) model to be the most effective. Although their study covered a larger geographic area, it mainly focused on classifying delays and faced challenges with imbalanced data and limited real-time data use. In a study conducted in 2024, Seongeun Kim and Eunil Park expanded the scope by applying a wider suite of ML models to predict departure delays at three major international airports [2]. Their models achieved high predictive accuracy, with rates of 74.9% for Incheon (ICN), 85.2% for John F. Kennedy (JFK), and 78.5% for Chicago Midway (MDW) in 2-hour forecasts. Although their study demonstrated the potential of ML models in long-term delay predictions, it was limited by its focus on individual airports and a reliance on historical datasets from 2011 to 2021, which may not fully capture future or emerging trends in weather patterns. In a similar study, Sun Choi and others also trained ML models with the goal of classifying whether a flight was delayed or on-time due to weather conditions. Their highest accuracy percentage was 80.36% using the Random Forest (RF) classifier [10]. Each of these studies focused on a binary classification model in which the aim was to classify a delayed or on-time flight due to various weather conditions without seeking to estimate the duration

of the delays. Though several studies have been conducted in which delay classification prediction was explored, the scope of the datasets and the use of regression models has been limited.

Building on these foundational studies, our research aims to address the limitations of previous work by integrating recent, high-frequency data across a more diverse range of U.S. airports. Unlike previous studies that focused on single airports or had limited data, we used advanced model selection and real-time data integration to improve prediction accuracy and generalizability. We compared the accuracy of classification and regression models in three scenarios: (1) when weather conditions aren't used as an independent variable, (2) when they're used in conjunction with flight data as independent variables, and (3) when weather conditions are used as an independent variable without flight data. By using this approach, we aim to determine if weather conditions are an effective predictor for precise models with a high level of accuracy in predicting flight delays. In developing models that account for the dynamic interactions between weather conditions and flight delays, we seek to provide stakeholders with actionable insights to reduce delay-related costs, environmental impacts, and improve overall passenger satisfaction. This research will not only advance the current state of delay prediction but also contribute to the broader field of transportation analytics, offering scalable solutions for mitigating weather-related disruptions across various modes of transportation. We believe that by using our diverse high-frequency dataset this study will find that weather conditions are an invaluable explanatory variable when it comes to classifying and predicting the duration of flight delays.

# 2 Sources

Having a valid and reliable source to find data for a ML project is essential. One of our main goals for this project is to determine the effectiveness of weather data as a predictor for flight delays. To carefully find the relationship between weather and flight delays, we needed a dataset that contained both the weather data and the flight data. However, after exploring online resources for such data, we found that this kind of dataset does not exist. The subsequent alternative to finding the data was to find one source for weather data and one for flight data, and then combine them.

We found the Department of Transportation (DOT) a credible source for flight data. The website allowed us to extract many important variables related to flight information such as flight date, time, airport identification number, departure time, and the number of minutes the flight is delayed (difference in minutes between scheduled and actual departure). The predictors from the flight data we used include variables such as month and year, origin and destination, time of day, airline, and airport. We created a binary column that indicates if a flight has been delayed for 5 minutes or longer which we'll use as the classification response variable. The column containing the number of minutes the flight has been delayed will be used as the response variable for our regression model.

Although numerous sources exist to find weather data such as Weather Underground and National Oceanic and Atmospheric Administration (NOAA), we found Iowa Environmental Mesonet (IEM) a reliable source for collecting the weather data. We found

this source credible for the following two reasons. First, IEM solely focuses on airport weather data, not only in the US but also at airports around the world, which can inform future work by applying the findings of this paper to global datasets. Secondly, IEM extracts the data through the Automated Surface Observing System and according to the IEM website, "ASOS networks are nationally monitored for quality 24 hours per day". This adds another level of confidence and surety to the validity and reliability of the data. The predictors from the weather data used include variables such as air temperature, dew point temperature, wind speed, wind direction, visibility, and pressure. We included weather and flight data of all the airports from 10 different states including: Florida, Texas, Colorado, New York, Illinois, California, Georgia, New Jersey, Maryland and Nevada between the years of 2014 and 2024.

After conducting a comprehensive literature review, we identified the key concerns and how different researchers approached weather-related flight delays. Almost all of the literature reviews have been published in the past 5 years which is very relevant to our research. The oldest one goes back to July 2007, which is still relevant today, but this paper only talks about the environmental impacts of flight delays. The studies that we have used as our sources have been published in credible journals in the fields of aviation, aerospace, and meteorology which are all essential fields of studies to understand the complex relationship between weather and flight delays. Not only that, most of our sources have been published by the most credible scientific publishers such as Springer and Cambridge University Press. These interdisciplinary publications provide us with a strong foundation on past work completed in this field and ensure the reliability and validity of our sources.

# 3 Exploration

This study involved collecting and integrating two primary datasets: weather data and flight data, spanning a decade and covering multiple airports across ten states. The data collection and cleaning processes were crucial in ensuring the datasets were accurate, aligned, and ready for analysis.

## 3.1 Data Collection

**Weather Data:**

The weather data was sourced from the Automated Surface Observing Systems (ASOS) and Automated Weather Observing Systems (AWOS), accessible via the Iowa Environmental Mesonet (IEM) ASOS Network. This dataset includes METAR-format observations from airport sensors worldwide.

Originally recorded at one-hour intervals, the data was accessed through a custom API request script. Records with missing or invalid temperature data were excluded, and to enable finer analysis, linear interpolation was applied to expand the data to 15-minute intervals.

**Flight Data:**

Flight data was manually collected from the Bureau of Transportation Statistics (BTS), Department of Transportation, using their online platform. Data for each month was downloaded in ZIP format, spanning the same ten states and years as the weather data.Then monthly files were decompressed and concatenated to create a comprehensive dataset.

## 3.2 Data Cleaning and Integration

**Alignment by Time and Location:**

The two datasets were merged based on datetime and airport station identifiers, linking flight records with corresponding weather data. The one-hour weather data intervals were interpolated to 15-minute intervals for precise temporal alignment. Additionally, the scheduled departure time in the flight data was rounded to the nearest 15 minutes to facilitate a left join with the weather data.

**Handling Missing Data:**

Missing values in weather data were addressed through forward-filling techniques, ensuring that essential values were available for each interval. Flight data underwent similar treatment, with forward-fill and linear interpolation applied to maintain consistency and prevent data gaps.

This systematic data collection, cleaning, and integration ensured robust alignment and quality, forming a reliable foundation for the subsequent analyses and model development. Table 1 presents the final set of variables included in our final dataset.

| Feature | Data Type | Description |
|---|---|---|
| Year | int64 | Year of the flight |
| Quarter | int64 | Quarter of the year |
| Month | int64 | Month of the flight |
| Day_of_Month | float64 | Day of the month |
| Day_of_Week | float64 | Day of the week |
| Operating_Carrier_Code | object | Code of the operating airline |
| Tail_Number | object | Aircraft tail number |
| Origin_Airport_ID | float64 | ID of the origin airport |
| Origin_Airport_Code | object | IATA code of the origin airport |
| Origin_State_Name | object | State name of the origin airport |
| Destination_Airport_Code | object | IATA code of the destination airport |
| Destination_State_Name | object | State name of the destination airport |
| Scheduled_Departure_Time | float64 | Scheduled departure time in minutes |
| Departure_Delay_Minutes | float64 | Departure delay in minutes |
| Taxi_Out_Time_Minutes | float64 | Taxi-out time in minutes |
| Flight_Distance_Miles | float64 | Distance of the flight in miles |
| Departure_Datetime | object | Exact departure datetime |
| Scheduled_Departure_Time_Minutes | float64 | Scheduled departure time in minutes |
| Air_Temperature_Fahrenheit | float64 | Air temperature at the origin airport |
| Dew_Point_Temperature_Fahrenheit | float64 | Dew point temperature at the origin airport |
| Relative_Humidity_Percent | float64 | Relative humidity at the origin airport |
| Wind_Direction_Degrees | float64 | Wind direction in degrees at the origin airport |
| Wind_Speed_Knots | float64 | Wind speed in knots at the origin airport |
| Hourly_Precipitation_Inches | float64 | Hourly precipitation in inches |
| Pressure_Altimeter_Inches | float64 | Altimeter pressure in inches |
| Sea_Level_Pressure_Millibar | float64 | Sea level pressure in millibars |
| Visibility_Miles | float64 | Visibility in miles |
| Sky_Cover_Level_1 | object | Sky cover description |
| Sky_Level_1_Altitude_Feet | float64 | Sky cover altitude in feet |
| Apparent_Temperature_Fahrenheit | float64 | Apparent temperature at the origin airport |
| Target | float64 | Target variable for prediction |

Table 1: Final Feature Set for Flight Delay Prediction Project

## 3.3 Data Balancing

**Initial distribution:**

After our data collection and cleaning process, the distribution of non-delayed (0) versus delayed (1) flights was as follows:

The minority (delayed flights) class making up only 25.18% of the dataset. This imbalance can affect a model's ability to accurately predict the occurrence of the minority class because it could be biased towards the majority class (not delayed flights).

**Balancing the data:**

To address the unbalanced issue of the two classes, we used Synthetic Minority Over-sampling Technique (SMOTE). This technique synthetically resamples the minority class in order to balance the dataset. The distribution of classes after we applied SMOTE is shown in figure 2. By using SMOTE, we were ensured that the models were no longer biased toward the majority class, which lead to more reliable and fair predictions.

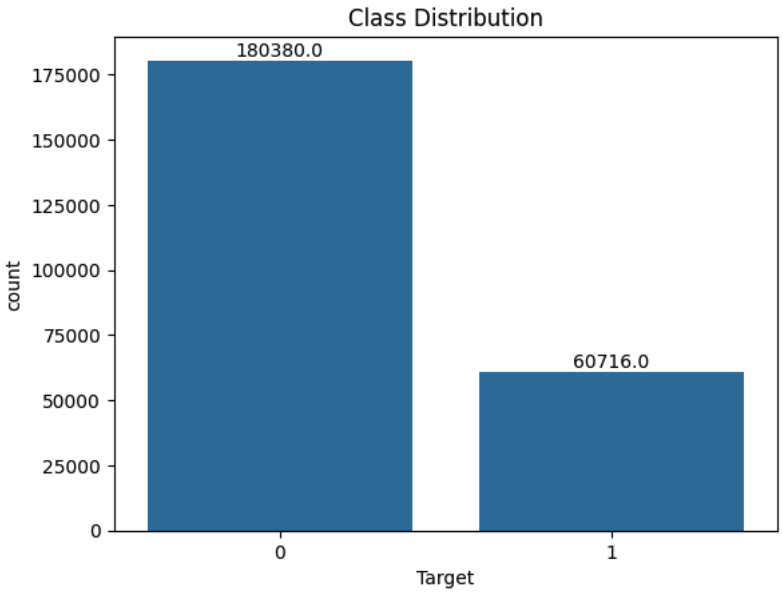



Figure 1: Class Distribution of the Dataset before SMOTE

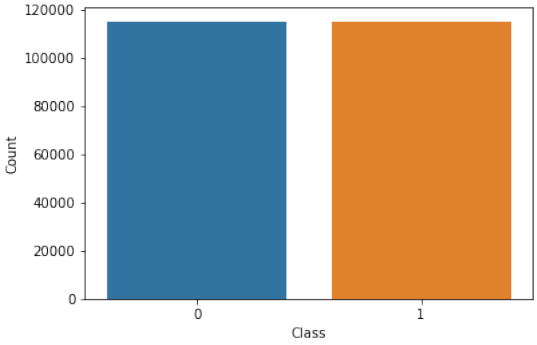



Figure 2: Class Distribution of the Dataset after SMOTE

# 4 Methodology

One of our research goals is to create a model that will accurately classify whether a flight delay will occur, so the methodology our project will employ is predictive. We will be using supervised statistical modeling approaches that are both parametric, such as logistic regression, and non-parametric, such as Support Vector Machines, Random Forest, Gradient Boosting Machine, Decision Trees, and CatBoosting. Exploring both parametric and non-parametric approaches will allow us to weigh the benefits and drawbacks of the interpretability, flexibility, and performance of a variety of models.

When using machine learning models, splitting your data into training, validation and testing is crucial in order to prevent overfitting. We used the common guideline suggested by data scientists to split our data into three sections. We allocated the data from years 2014-2019 (70 % of the data) to the training set, the data from years 2020-2022 (10 % of the data) to the validation set, and the data from years 2023-2024 to the testing set (20 % of the data).

**Classification models**:
Our goal was to classify whether a flight will be delayed or not. For this case, we leveraged

the following classification models using Python:

## 4.1 Logistic Regression (LR) Classification

A LR model performs a binary classification by predicting which class the data has a higher probability of belonging to. We first leveraged an intercept-only LR model that estimated the probability of a flight delay based solely on the overall class distribution, not taking into account any predictors. By doing so, we established an accuracy level to compare the performance of our more complex classification models to. Additionally, we fit a full LR model with predictors to classify flight delays. This method of parametric modeling assumes a linear relationship between the predictors and the log odds of the outcome, and is therefore more easily interpretable. That said, LR is more prone to bias due to its sensitivity to anomalies and its highly generalizable nature.

## 4.2 Linear Support Vector Classification (LinearSVC)

With millions of observations in our dataset, we selected Linear Support Vector Classification (LinearSVC) instead of the standard SVC model due to its efficiency in both speed and memory usage, which makes it much more suitable for handling large datasets. LinearSVC works by finding the most effective dividing line, or "hyperplane," that separates delayed flights from non-delayed ones, aiming to maximize the distance between this boundary and the closest data points. This approach helps enhance the model's accuracy and stability. Unlike standard SVC, which uses a more complex optimization process, LinearSVC focuses on a simpler objective function, making it faster to compute on larger datasets. Additionally, LinearSVC provides interpretable results by highlighting which features have the greatest influence on predicting flight delays, offering valuable insights into the factors that contribute to these delays.

## 4.3 Decision Tree (DT)

Another method used to classify a flight was delayed or not was using a Decision Tree (DT). DT is a supervised machine learning algorithm that is used for both classification and regression. Kim and Park [2] recommends this method due to its versatility and interpretability. The resulting model, has a tree structure, where each node represents a decision based on a particular feature from the data, and each branch represents the outcome of that. The tree starts from the first node also known as the root node, and recursively splits into further nodes based on the data. The recursive process is continued until the model reaches the leaf node which corresponds to the final prediction.

A common issue with DT is overfitting. We used grid search to test and tune different hyperparameters in order to avoid overfitting and optimize its performance.

## 4.4 Random Forest (RF)

The RF model combines the outputs of multiple DT models and leverages each of their results to perform both classification and regression predictions. This method of machine learning randomly selects a subset of features to build each decision tree; a final prediction will be determined by majority vote of all of the decision trees' predictions. By averaging predictions across multiple decision trees formulated by randomized features, variance is reduced and accuracy is often improved when compared to DT models. This is advantageous when utilizing large datasets with high-dimensional features.

## 4.5 CatBoost Classification (CatBoost)

CatBoost, a gradient boosting algorithm optimized for handling categorical features, and it was chosen to classify flight delays due to its capability to process high-dimensional data without extensive preprocessing. Unlike traditional boosting methods, CatBoost implements ordered boosting, which reduces prediction bias and mitigates overfitting by training trees sequentially in a way that prevents data leakage. The algorithm builds multiple decision trees that work together, with each tree correcting errors from previous trees, resulting in improved predictive accuracy. Hyperparameters such as the learning rate, tree depth, and L2 regularization were carefully tuned to achieve an optimal balance between model complexity and accuracy. Additionally, class weights were applied to address the imbalance between delayed and non-delayed flights, allowing CatBoost to learn effectively from both classes. By setting early stopping rounds, we monitored performance on the validation set to further prevent overfitting. This makes CatBoost particularly suited for high-cardinality features and imbalanced datasets, offering a powerful, interpretable solution for classification tasks.

# 5 Results

In this section we discuss the result of our analysis by leveraging machine learning models mentioned in 3.1 (Classification models). Table 1, shows the performance on the training data, Table 2 provides the metrics used to evaluate the model on the validation set and Table 3 provides the metrics used to evaluate the model performance on the unseen or real-world data, the testing set.

| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Baseline | 0.75 | 0.59 | 0.33 | 0.42 |
| LR | 0.67 | 0.43 | 0.64 | 0.51 |
| Linear SVC | 0.69 | 0.69 | 0.69 | 0.68 |
| DT | 1 | 1 | 1 | 1 |
| RF | 1 | 1 | 1 | 1 |
| CatBoost | 0.76 | 0.54 | 0.71 | 0.61 |

Table 2: Performance metrics on the training set

| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Baseline | 0.80 | 0.39 | 0.19 | 0.26 |
| LR | 0.67 | 0.28 | 0.51 | 0.36 |
| Linear SVC | 0.80 | 0.61 | 0.56 | 0.56 |
| DT | 0.60 | 0.75 | 0.60 | 0.65 |
| RF | 0.77 | 0.34 | 0.28 | 0.31 |
| CatBoost | 0.69 | 0.30 | 0.50 | 0.37 |

Table 3: Performance metrics on the validation set

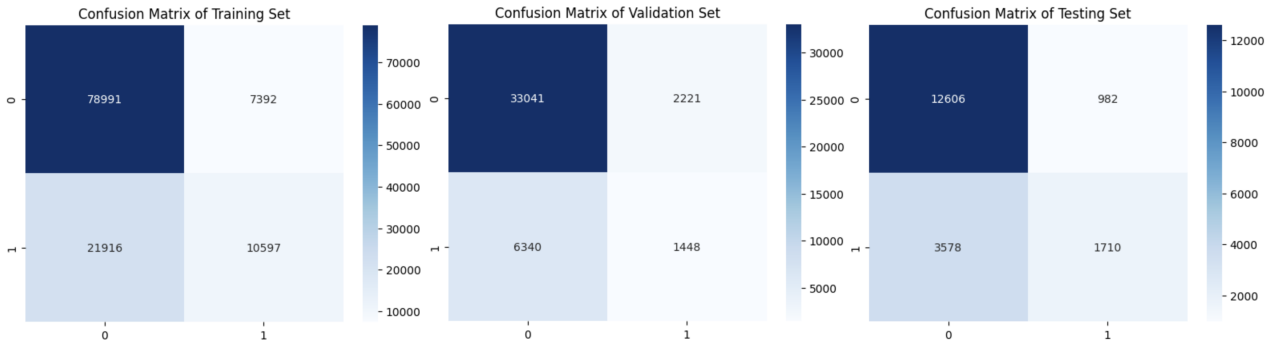
| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Baseline | 0.76 | 0.64 | 0.32 | 0.43 |
| LR | 0.67 | 0.44 | 0.67 | 0.51 |
| Linear SVC | 0.74 | 0.70 | 0.56 | 0.55 |
| DT | 0.57 | 0.69 | 0.57 | 0.59 |
| RF | 0.73 | 0.54 | 0.35 | 0.42 |
| CatBoost | 0.66 | 0.42 | 0.53 | 0.47 |

Table 4: Performance metrics on the testing set

- Logistic Regression (LR) Classification:

First, we fit the unbalanced data to an intercept-only LR model in order to establish a baseline measurement for prediction accuracy to compare our subsequent models to. After fitting this model on the training data the model had an overall validation accuracy of 0.81 and test accuracy of 0.76. Since our study is focused on classifying delayed flights, we decided to examine the precision, recall, and F1 scores for each model. High precision would indicate that when the model predicts a delayed flight it is often correct, and high recall would indicate that the model correctly identifies most delayed flights in our dataset. A high F1 score is the harmonic mean of precision and recall, so a high F1 score

would indicate that the model has a good balance between precision and recall. Since our dataset is unbalanced, the baseline models have a relatively high accuracy level because a 50/50 prediction will correctly classify the majority class more often. In Figure 3 you can see the confusion matrices for the training, testing, and validation sets for our baseline model, showing the quantity of instances in each class and whether it was correctly predicted.




(a) Training Set (b) Validation Set (c) Test set

Figure 3: Confusion Matrix for training, validation and testing set

The ROC curve for our Baseline model can be seen below:

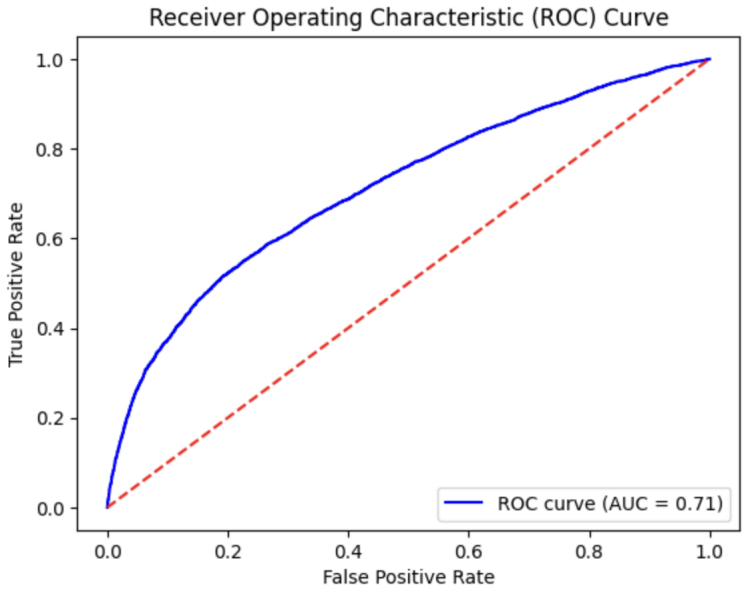
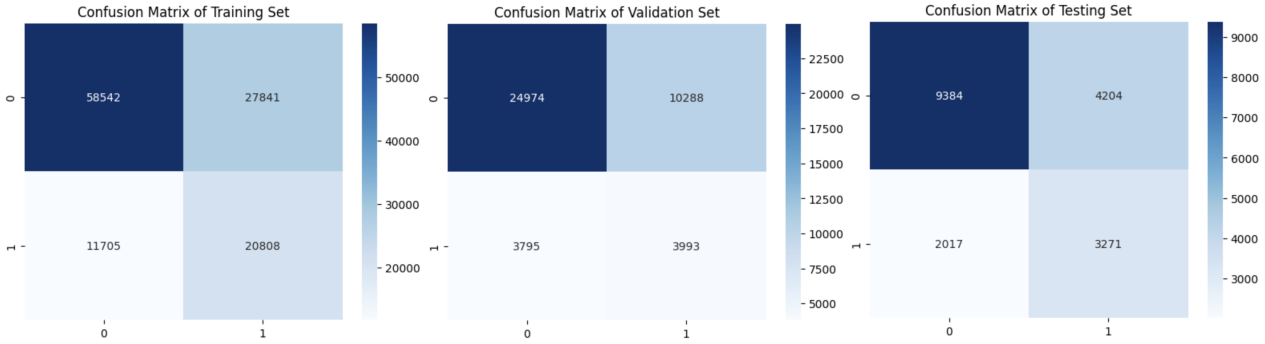



Figure 4: Baseline LR ROC Curve

Next, we fit a full LR model to evaluate the effectiveness of a parametric model that assumes a linear relationship between the weather predictors and the occurrence of flight delays. Comparing the overall accuracy, precision, recall, and F1 scores of the full LR

model to the baseline model, the scores are only marginally better in some cases and worse in others. This would suggest that weather predictors do not improve performance accuracy, or this could be indicative of a non-linear relationship between the predictors and the occurrence of delayed flights. See the confusion matrices for the full LR model in Figure 5.




(a) Training Set (b) Validation Set (c) Test set

Figure 5: Confusion Matrix for training, validation and testing set

A ROC curve (based on the test set) is provided below in order to visualize the performance of the model's ability to accurately predict flight delays:

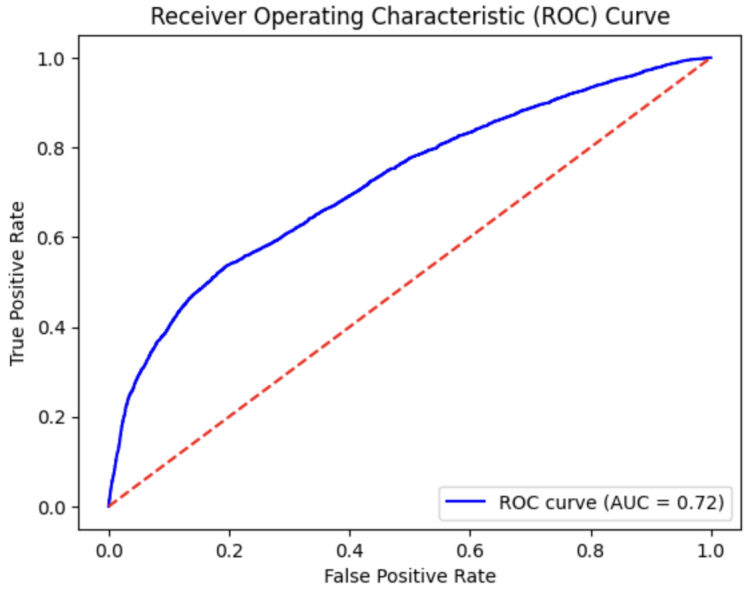


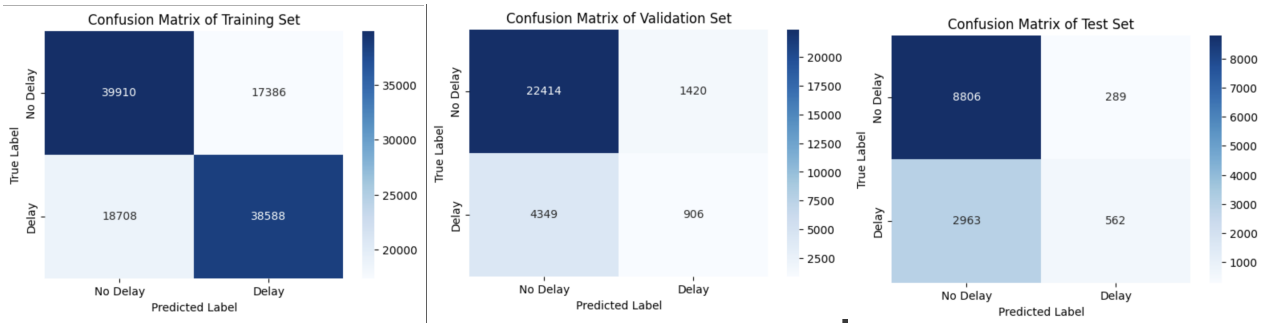

Figure 6: Full LR ROC Curve

As you can see, the trade-off between true positive classifications over false positive classifications increases sharply in the beginning, but the inclusion of more false positives with additional true positives as the rates increase suggest a decrease in discriminative power.

We now move on to our non-parametric classification models.

• Linear Support Vector Classification (LinearSVC):

The preliminary results of our model show moderate success in distinguishing between delayed and non-delayed flights, with a validation accuracy of 80 percent and a test accuracy of 74 percent. However, accuracy alone does not fully capture the model's limitations, especially in identifying delayed flights. While the model performs well for non-delayed flights, achieving high precision and recall for this majority class, it struggles with the minority class of delayed flights. This is evident in the low recall and F1-score for delayed flights, suggesting that the model frequently misses these cases. Here we provide the confusion matrices for all training, testing, and validation sets:




(a) Training Set (b) Validation Set (c) Test set

Figure 7: Confusion Matrix for training, validation and testing set

The confusion matrices show that the model is good at predicting flights that won't be delayed across all sets, but it struggles to accurately identify delayed flights, especially as it moves from training to validation to test data. This trend suggests that while the model learns well from the training data, it has difficulty generalizing, particularly for predicting delays, indicating it could benefit from adjustments that are listened in a future paragraph to improve accuracy on real-world data.

Now lets look at feature importance to see which variables were most influential to our model:

The feature importance results show which factors most influence the model in predicting flight delays, based on the training data. Key factors like dew point temperature, scheduled departure time, and humidity stand out, suggesting weather and timing significantly impact delays. We used the training set for this analysis to ensure that the model's learning process is separate from the validation and test sets, helping us avoid data leakage. This approach allows us to see which features the model relies on to make predictions without impacting its performance on new, unseen data.

Moving on, lets explore the results of the ROC Curve (based on the test set):

The ROC curve shows that the model has moderate discriminatory power with an Area Under the Curve (AUC) of 0.70. This indicates that the model can distinguish between delayed and non-delayed flights better than random guessing (an AUC of 0.5) but still has room for improvement. The curve's gradual rise suggests that the model achieves a decent true positive rate but at the cost of a higher false positive rate, especially as it tries

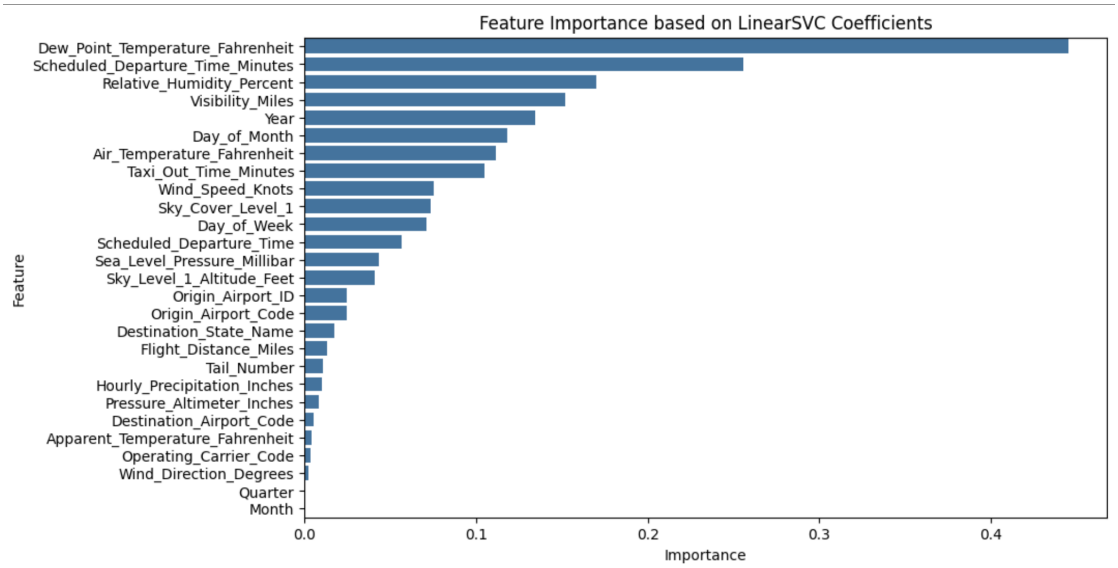



Figure 8: LinearSVC Feature Importance

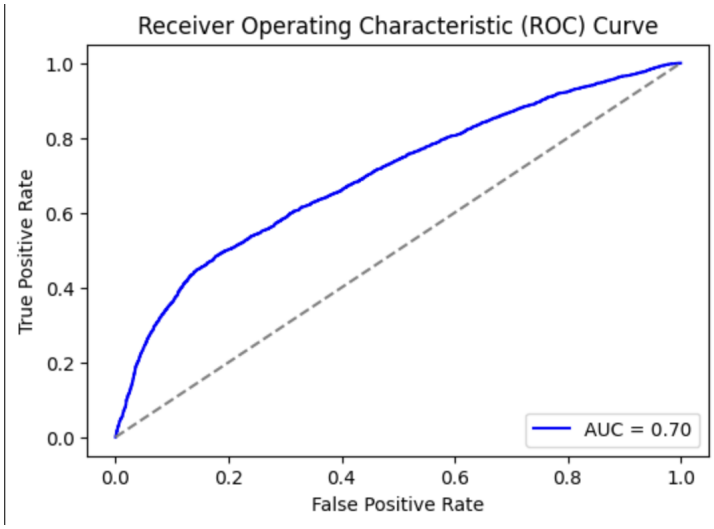


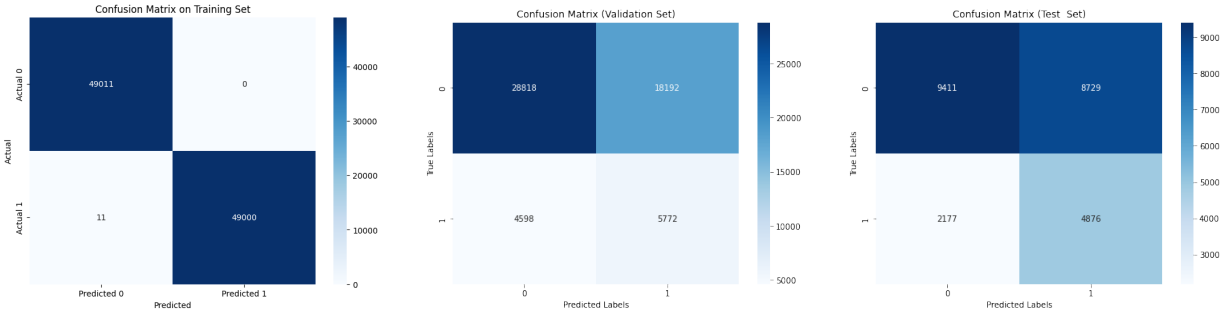

Figure 9: LinearSVC ROC Curve

to capture more delayed flights. Overall, while the model shows some ability to predict delays, further optimization is needed to improve its reliability.

Although SMOTE was used to balance the dataset during training, issues with recall for delays persist. This indicates potential for improvement, and additional techniques, such as experimenting with alternative sampling methods (like ADASYN or Borderline-SMOTE), adjusting the decision threshold, or exploring cost-sensitive learning, could enhance the model's performance. These results suggest that the selected methodology has the potential to address the research question of predicting flight delays but may need further refinement to achieve reliable results for real-world application. We are optimistic about improving recall and achieving a better balance across classes through these next steps, which will help ensure the model's predictions are both actionable and accurate.

- Decision Tree (DT):

The decision tree model appears to be overfitting the training data, achieving perfect scores on the training dataset. This may be due to the dataset imbalance, which was addressed using SMOTE.




(a) Figure 1 (b) Figure 2 (c) Figure 3

Figure 10: Confusion Matrix for training, validation and testing set

The overfitting is most obvious in Figure 1(a). There are 0 false positives, while the count of false negatives is 11; hence, this model is almost perfect on a training set. However, when fitting the model to the validation set that contained unbalanced data, the false positives went as high as 18192 and the number of false negatives as high as 4598. Such an increase already hints at poor generalization of the model beyond the training data.

This translates into 8729 false positives and 2177 false negatives when applied to unseen, real-world data or the testing set; that is, it does better but still can make mistakes. Fitting to unseen data decreases the accuracy of the model; but this is to be expected, however this poor performance on testing shows that the model is now less overfitted and can generalize better on the real-world data.

The reason that the model was less over fitted in the validation and testing set was because we used Grid Search [10] to find the best parameters. Table 5, shows the different parameter used to find the best and optimal tree.

| **Hyperparameter** | **Values** |
|---|---|
| `max_leaf_nodes` | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 50, 100 |
| `min_impurity_decrease` | 0.0001, 0.0005, 0.001, 0.005, 0.01 |
| `max_depth` | 1, 3, 5 |
| `min_samples_split` | 2, 3, 4, 5, 6 |

Table 5: Hyperparameter Grid Search Values

After using the Grid Search to find the optimal based on the hyperparameter

values from table 5, we found the following values to result in the most optimal tree shwon in table 6. Lastly, we were also interested in which variables were

| Hyperparameter | Optimal Value |
|---|---|
| `max_leaf_nodes` | 6 |
| `min_impurity_decrease` | 0.001 |
| `max_depth` | 5 |
| `min_samples_split` | 4 |

Table 6: Optimal Hyperparameter Values for Decision Tree

the most important for classifying flight delays based on the decision tree. We used the feature importance from the decision tree to plot the most important features based on the decision tree model as shown in Figure 5.

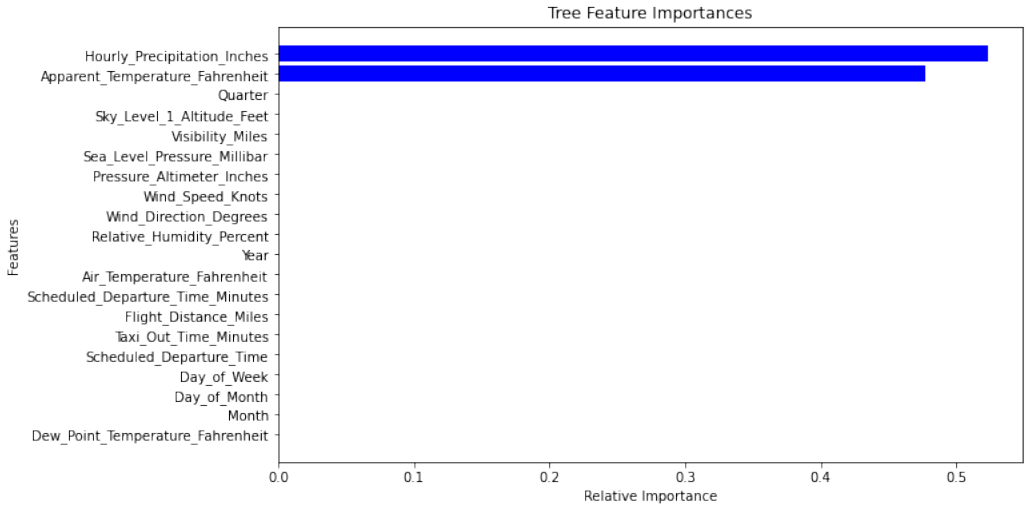
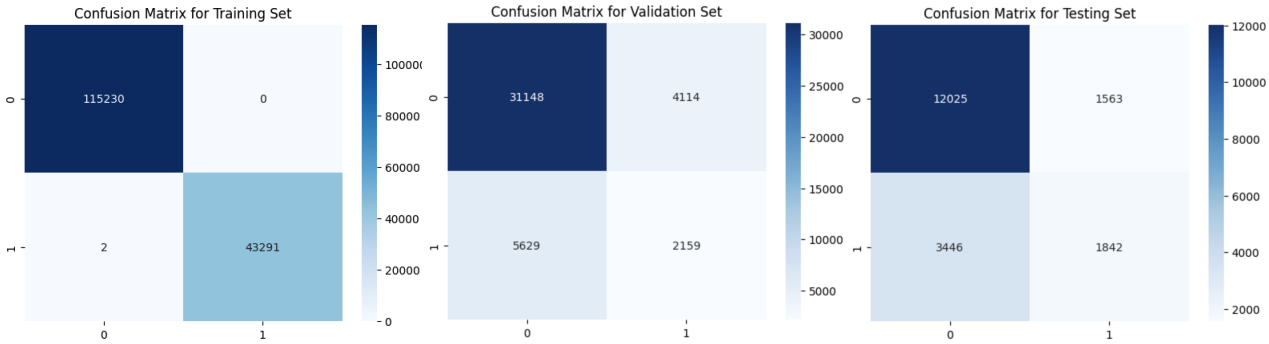



Figure 11: Tree Feature Importance

Based on Figure 5, the most important features according to the decision tree model are hourly precipitation in inches and apparent temperature in Fahrenheit.

- Random Forest (RF):

After fitting a RF model to the training set, the validation set received an overall accuracy score of 0.83, and the testing set received an overall accuracy score of 0.73. The confusion matrices for which can be seen below:




(a) Figure 1 (b) Figure 2 (c) Figure 3

Figure 12: Confusion Matrix for training, validation and testing set

Since RF models leverage multiple decision trees in order to perform a more accurate and less overfit classification, we would expect our accuracy measures to improve with the RF fit. Although the overall accuracy improved compared to the DT model, the measures of precision, recall, and the F1 score performed consistently worse.

The ROC curve for our RF model can be seen below:

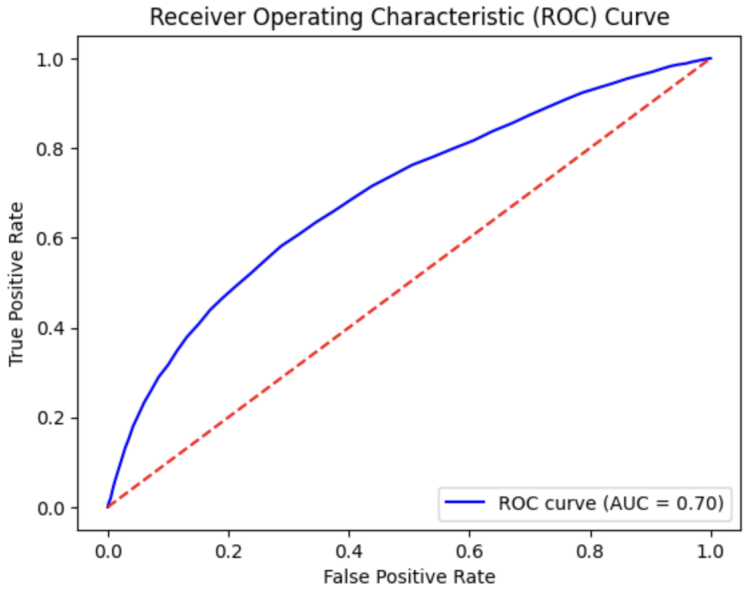



Figure 13: RF ROC Curve

- CatBoost Classification (CatBoost):

CatBoost, a gradient boosting algorithm optimized for handling categorical variables efficiently, was chosen for predicting flight delays due to its ability to process high-dimensional data without extensive preprocessing. Unlike traditional boosting methods, CatBoost incorporates ordered boosting, which mitigates overfitting and reduces prediction bias. Key hyperparameters such as the learning rate (0.05), tree depth (8), and L2 regularization (2) were tuned to balance model complexity and accuracy. To address the class imbalance between delayed and non-delayed flights, we applied class weights, enhancing the model's ability to learn from the minority class. Early stopping rounds were set to 200, ensuring the model did not overfit on the validation set. CatBoost achieved strong performance metrics, with an F1 score of 0.512 on the validation set, showing its capability to capture complex patterns in flight delay prediction. Its interpretability and efficiency in processing categorical data make CatBoost a robust choice for classification in this high-dimensional context.

# 6 References

[1] Yhdego et al. “Analyzing the Impacts of Inbound Flight Delay Trends on Departure Delays Due to Connection Passengers Using a Hybrid RNN Model.” Mathematics 2023, 11, 2427.

[2] Kim, S., Park, E. Prediction of flight departure delays caused by weather conditions adopting data-driven approaches. J Big Data 11, 11 (2024).

[3] Goodman, C. J., and J. D. Small Griswold, 2019: Meteorological Impacts on Commercial Aviation Delays and Cancellations in the Continental United States. J. Appl. Meteor. Climatol., 58, 479–494

[4] Kiliç, K.; Sallan, J.M. Study of Delay Prediction in the US Airport Network. Aerospace 2023, 10, 342

[5] Gratton, G. B. et al. “Reviewing the Impacts of Climate Change on Air Transport Operations.” The Aeronautical Journal 126.1295 (2022): 209–221. Web.

[6] Tileagă, C., Oprișan, O. (2021). Flights Delay Compensation 261/2004: A Challenge for Airline Companies?. In: Orăștean, R., Ogrean, C., Mărginean, S.C. (eds) Organizations and Performance in a Complex World. IECS 2019. Springer Proceedings in Business and Economics. Springer, Cham.

[7] Carlier, Sandrine, Ivan de Lépinay, Jean-Claude Hustache, and Frank Jelinek. Environmental Impact of Air Traffic Flow Management Delays. CiteSeerX, doi:10.1.1.76.3545.

[8]Sekelová, I., Korba, P., Pjurová, S., Marimuthu, S., Kale, U. (2023). Reducing the Environmental Impact of Aviation by Minimizing Flight Delays. In: Sogut, M.Z., Karakoc, T.H., Secgin, O., Dalkiran, A. (eds) Proceedings of the 2022 International Symposium on Energy Management and Sustainability . ISEMAS 2022. Springer Proceedings in Energy. Springer, Cham.

[9] Hsu et al. “Unraveling Extreme Weather Impacts on Air Transportation and Passenger Delays using Location-based Data.”

[10] “Prediction of Weather-induced Airline Delays Based on Machine Learning Algorithms.” 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), Digital Avionics Systems Conference (DASC), 2016 IEEE/AIAA 35th.